Integration of a heterogeneous compute resource in the ATLAS workflow

Felix Bührer[1], *Anton* Gamel[1,2,*], *Benoit* Roland[1], *Ulrike* Schnoor[1,3], and *Markus* Schumacher[1] on behalf of the ATLAS Collaboration

[1]Institute of Physics, University of Freiburg, Freiburg, Germany
[2]Computing Center, University of Freiburg, Freiburg, Germany
[3]now at CERN, Geneva, Switzerland

Abstract. With the ever growing amount of data collected by the experiments at the Large Hadron Collider (LHC), the need for computing resources that can handle the analysis of this data is also rapidly increasing. This increase will only be amplified after upgrading to the High Luminosity LHC [1]. High-Performance Computing (HPC) and other cluster computing resources provided by universities can be useful supplements to the ATLAS collaboration's own WLCG resources for data analysis and production of simulated event samples. The shared HPC cluster "NEMO" at the University of Freiburg has been made available to local ATLAS users through the provisioning of virtual machines incorporating the ATLAS software environment analogously to the bare metal system of the local ATLAS Tier2/Tier3 centre. In addition to the provisioning of the virtual environment, the on-demand integration of these resources into the Tier3 scheduler in a dynamic way is described. Resources are scheduled using an intermediate layer, monitoring requirements and requesting the needed resources.

ATL-SOFT-PROC-2018-025
24 November 2018

1 Introduction

The analysis of collision data collected at the LHC and simulation of events is primarily done at 2 Tier0, 11 Tier1 and 160 Tier2 sites within the WLCG [2]. Tier3 components at sites provide resources for local groups. WLCG clusters are mostly set up for High Throughput Computing (HTC) to get as much compute power as possible. High Performance Computing (HPC) clusters, as provided by universities and other institutions, sometimes even co-located at the same sites, may be used for HTC-like workflows to extend the capacities of the existing WLCG resources. However, there are no standards interfaces or abstractions layers available to easily cross-link clusters with different HPC/HTC set-ups, different resource managers or different login schemes. To configure a cluser to accept workloads from secondary schedulers or to deliver basic monitoring information, that can be passed through, is not a straight-forward task.
In order to achieve the on-demand scheduling of resources on one cluster due to requirements on a different cluster, an intermediate layer, or meta-scheduler, is put in place.

*e-mail: anton.gamel@physik.uni-freiburg.de

From its primary concept, the NEMO HPC cluster was designed to provide a full virtualization solution with OpenStack [3] that enables users to spawn virtual machines (VMs) with a pre-configured image - so-called virtual research environments (VREs) [4]. These VMs can be requested like any other user job via the NEMO batch system. After start-up they can be used to run jobs that require the full ATLAS software environment [5].

In the following, we describe how these virtual resources are integrated into the ATLAS Tier2/Tier3 cluster. NEMO and the Tier2/Tier3 are utilizing different batch systems. On the frontend side, SLURM [6] is used as a scheduler, while the NEMO cluster backend runs a combination of MOAB & Torque [7]. We use ROCED [8], developed at the Karlsruhe Institute of Technology (KIT) as meta-scheduler. ROCED is used already to integrate NEMO resources into the HTCondor [9] system at KIT. Due to the modular architecture of ROCED, it can also be used for connecting the two systems in Freiburg. To do so, a new component monitoring the SLURM queue on the frontend has been developed.

The performance of the system is being measured using several different benchmarks. These benchmarks are also used to quantify the modification of the performance due to changes in the configuration of the meta-scheduler and of the virtual machines being spawned. They will also be part of a future continuous monitoring effort in order to be able to detect changes in the submitted workloads. This monitoring and tuning effort will ensure a stable but also dynamic and efficient environment.

2 Challenges

The ATLAS research groups in Freiburg have very specific requirements to the operating system as well as the installed software. The reason for this is that ATLAS software must be able to run on all grid sites of the WLCG.
Virtualization has been found to be a technology that can simplify the challenge to provide a stable environment on a range of different heterogeneous and changing platforms, especially in the context of particle physics [10].
Being only one of multiple user groups on a shared HPC system, especially the choice of operating system has to take into account considerations from all user groups as well as from the party operating the cluster. A fully virtualized environment, independent of the choices made on the HPC cluster itself, will give the best possible scope to implement a system, that looks and behaves in exactly the same way as the non-virtualized Tier2/Tier3 cluster. This consistency between the two systems would also make it possible in the future to reroute ATLAS production jobs to either NEMO or any other opportunistic resource that is able to run the provided VM image without any additional work. The VM images which are made available to OpenStack on NEMO have to be created and updated easily in an automatic procedure and have to fulfil the following requirements:

- Scientific Linux 6 [11] - current OS on the Tier2/Tier3 cluster
- Access to ATLAS software via the CERN virtual file system CVMFS [12]
- Tier3 user access scheme and environment
- Access to both grid-aware datasets on the distributed storage system dCache [13] and the local NEMO parallel filesystem BeeGFS [14].

Since the VMs are completely self-contained, all features needed to monitor and benchmark the machine are independent of the two schedulers that are involved and can either be implemented on the VM itself or offloaded to the meta-scheduler. In the future, these

information will also be used for continuous monitoring of the robustness and performance of the system.

3 Installation

3.1 Generation of the virtual machines

The VM template is generated with Packer [15] using a Scientific Linux 6 netinstall image as base. The customization and configuration of the template is done with puppet [16] which is also used for the contextualisation of the non-virtualized worker nodes on the Tier2/Tier3 cluster. Changes in configuration are automatically picked up by both systems. The output of this procedure is a static image that can be uploaded to the OpenStack server and is directly available.
The VMs do not contain a predefined CVMFS cache and do not utilize the hard disk as persistent cache. A 256 MB RAM disk is being filled on demand from the local frontier squid [17] proxies. This circumvents an on-disk CVMFS installation on the host. This is a different approach to other solutions relying on access to software from CVMFS and using an on-disk cache like CernVM [18].

3.2 Connection of front and backend batch systems

The biggest challenge for a smooth operation is the interconnection between the two different batch systems – SLURM on the Tier2/Tier3 frontend and MOAB on the backend (NEMO) through a meta-scheduler.
The workflow is as follows: The meta-scheduler monitors the frontend scheduler to which users send their workload. It then maps the requirements to the available configurations on the NEMO HPC, and decides on the number of batch jobs each spawning a virtual machine with the given configuration on the backend batch system. The jobs that are used to start VMs in the OpenStack environment are regular user jobs on NEMO, that are started according to the availability of resources and the fairshare of the user. When the VMs start, they need to be integrated as additional resources into the SLURM scheduler in order for the user jobs to be allocated to these resources.

Figure 1 shows the general mode of operation. For now, SLURM in the Tier2/Tier3 cluster is set-up with separate partitions that reflect the user's affiliation to one of the local ATLAS working groups. Each working group is in turn represented by a single user on NEMO, that is used to queue the jobs starting the VMs. By this mechanism, the fair share for the different areas of research using NEMO are incorporated into the workflow.
ROCED monitors these partitions. After user's job submission ROCED starts a wrapper job on the backend batch system, mapped to a single dedicated user per group. The wrapper job then triggers the start of a VM with a given configuration on OpenStack. As long as resources are requested and available on NEMO, additional virtual machines can be started. This mechanism leads to a dynamic extension of the amount of compute nodes and job slots available for physics analyses on the frontend system.
Before VMs are integrated into SLURM, a diagnosis-check is done to see whether all needed resources are available. After a successful check the VM is set to online in SLURM and jobs can be allocated.

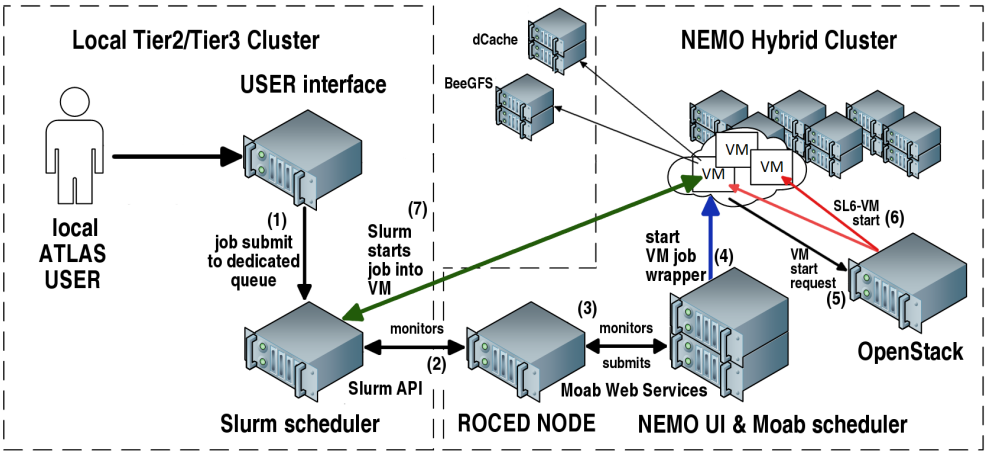

Figure 1. Schematic view of how user jobs submitted to the frontend scheduler SLURM trigger jobs to start VMs on the OpenStack instance at NEMO.

4 Benchmarks

In addition to the legacy HEP-SPEC06 (HS06) benchmark [19], the evaluation of the performance of the compute resources makes use of three benchmarking programs available in the CERN benchmark suite [20]:

1. Dirac Benchmark 2012 DB12 [21]
2. Whetstone benchmark [22]
3. Kit Validation benchmark KV [23].

The DB12 and Whetstone programs are evaluating the performance of CPUs through floating-point arithmetic operations.
One of the main differences between the two benchmarks resides in the variables used as input to the arithmetic operations: DB12 is using random numbers generated according to a Gaussian distribution, while Whetstone is using variables with predefined values.
The KV benchmark is making use of the ATLAS software ATHENA[24] to simulate and reconstruct the interactions of muons in the detector of the ATLAS experiment.

As our primary target is to measure performances of CPUs in the context of High Energy Physics (HEP) applications, the KV benchmark constitutes a realistic payload, more suited to our goal than the DB12 and Whetstone software. The DB12 benchmark is using the HS06 units, Whetstone scores are expressed in Million of Whetstone Instructions Per Second (MWIPS), and the KV output provides the number of events produced per second. The different benchmarks are used to evaluate the performance of identical 20 cores Intel Xeon E5-2630 CPUs on the two different clusters: the Tier2/Tier3 cluster (ATLAS-BFG) and the shared HPC cluster (NEMO). The performance has been evaluated on three different configurations; the Tier2/Tier3 and NEMO HPC clusters running both on bare metal and the virtual machines running on the NEMO HPC cluster. On the Tier2/Tier3 bare metal and on the VMs running on the NEMO cluster, hyper-threading (HT) technology is activated. Both are using Scientific Linux 6 as operating system. The NEMO bare metal has no HT activated due to the

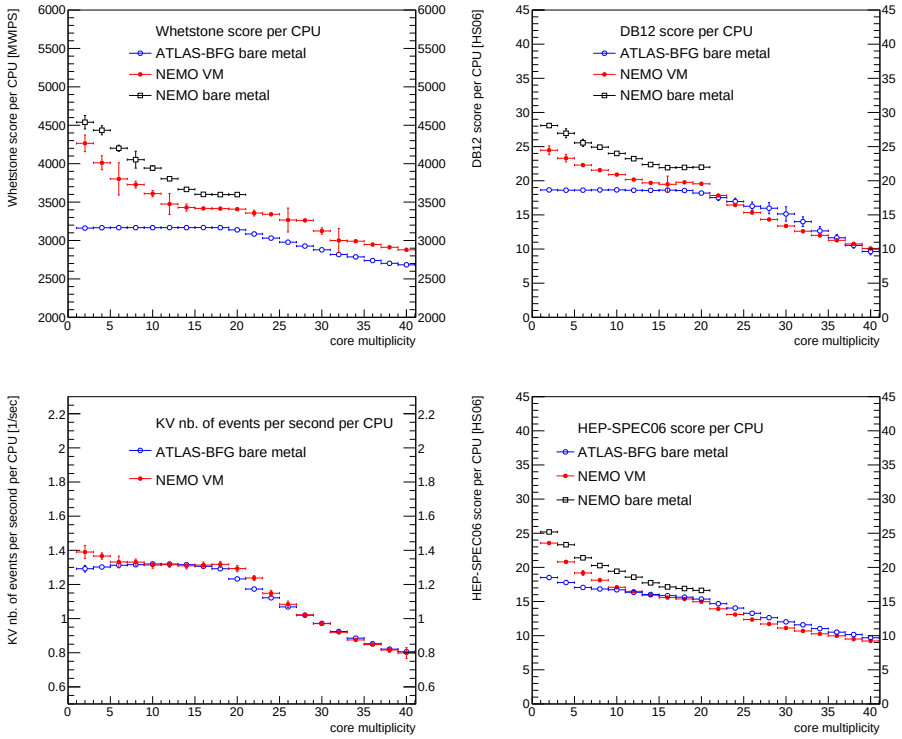

Figure 2. Score per CPU as a function of the core multiplicity for the Whetstone (top left), DB12 (top right), KV (bottom left) and HEP-SPEC06 (bottom right) benchmarks for the Tier2/Tier3 (ATLAS-BFG) running bare metal (blue open circles), the NEMO VMs (red full circles) and the NEMO bare metal (black open squares). The data points represent the average values of the benchmarks for each core multiplicity, and the vertical bars show the associated standard deviations.

more general use case of the system, and uses CentOS7[25] as operating system. The scores of the HEP-SPEC06, DB12, Whetstone and KV benchmarks have been determined for these three configurations as a function of the number of cores actually used by the benchmarking processes. This number ranges from 2 to 40 for the Tier2/Tier3 bare metal and for the VMs running on the NEMO cluster, for which HT is enabled, and from 2 to 20 for the NEMO bare metal, for which HT is not implemented. The results have been determined by step of two core units. The benchmarks have been run 20 times for each core multiplicity value, and the means and standard deviations of the corresponding distributions have been extracted.

The scores per CPU and the total scores are presented in figures 2 and 3 respectively, for the four benchmarks and the three configurations considered, except for the KV software for which the NEMO bare metal results are not yet available. This latter benchmark retrieves the ATLAS software from the CVMFS file system, which is presently not available for the NEMO bare metal.

A decrease of the Whetstone, DB12 and HEP-SPEC06 scores per CPU is observed in figure 2 for increasing values of the core multiplicity for the NEMO VMs and NEMO bare metal configurations. For the Tier2/Tier3 bare metal, the scores per CPU remain

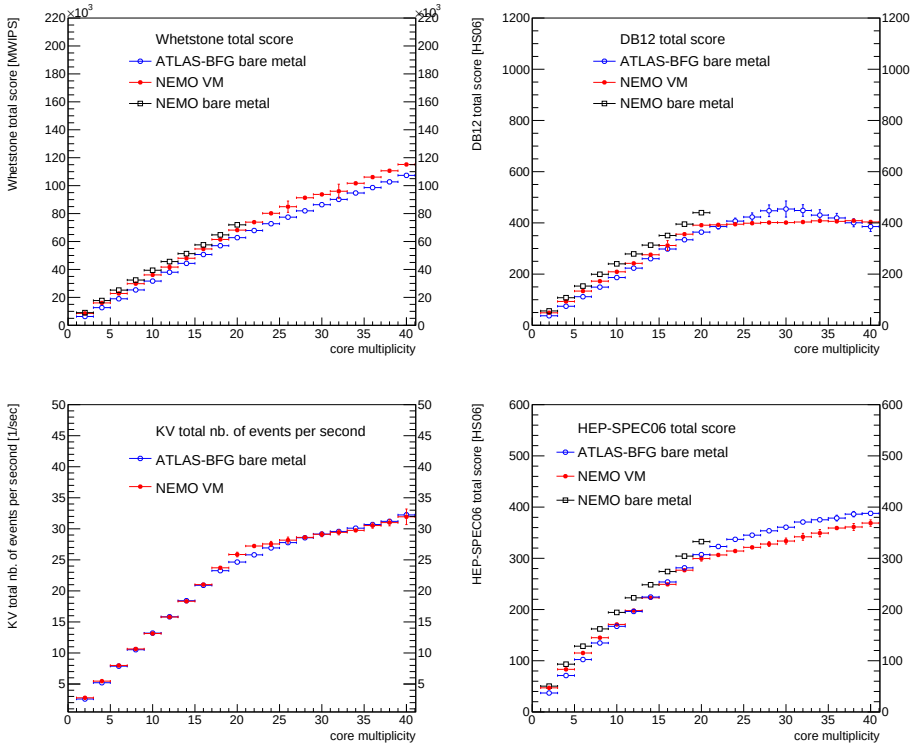

Figure 3. Total score as a function of the core multiplicity for the Whetstone (top left), DB12 (top right), KV (bottom left) and HEP-SPEC06 (bottom right) benchmarks for the Tier2/Tier3 (ATLAS-BFG) running bare metal (blue open circles), the NEMO VMs (red full circles) and the NEMO bare metal (black open squares). The data points represent the average values of the benchmarks for each core multiplicity, and the vertical bars show the associated standard deviations.

constant until the maximum number of physical cores is reached, and only start to decrease in the region where hyper-threading is active. The KV results exhibit a similar behaviour for both the Tier2/Tier3 bare metal and the NEMO VMs, with a constant number of events produced per second per CPU below the maximum number of physical cores and a decrease of the performance afterwards. This behaviour is the closest to the pattern expected for an ideal CPU benchmark: a constant CPU performance per physical core and a decrease in the region where HT is active. The Whetstone score per CPU at a core multiplicity of 20, the maximum number of physical cores available, is considered as an illustrative example of the benchmark behaviours on the three different configurations. An increase of the CPU performance by the order of 5% is observed when going from the Tier2/Tier3 bare metal to the NEMO VMs, while going from the NEMO VMs to the NEMO bare metal leads to a further increase of performance of the order of 5% as well.

A continuously increasing total score is observed in figure 3 for the Whetstone benchmark on the three different configurations, while the DB12, KV and HEP-SPEC06 results are characterized by a flattening increase or a constant behaviour once the maximum number of physical cores has been reached. The Whetstone benchmark provides higher CPU

performances in the HT region in comparison to the scores obtained with the three other benchmarks. The scores obtained with the KV and HEP-SPEC06 benchmarks indicate an increase of the CPU performance by 15 to 20% when going from the maximum number of physical cores to the upper edge of the HT region, while the Whetstone scores exhibit a larger increase of the order of 60%. The Tier2/Tier3 bare metal and the VMs running on the NEMO cluster share the same configuration in terms of hardware, operating system and hyper-threading. A given benchmark should therefore exhibit a similar behaviour for both configurations. The KV benchmark, besides being the more realistic estimator of the CPU performance in the context of High Energy Physics applications, is the only benchmark for which this expectation is observed. The behaviours of the different benchmarks still need to be studied in more details, in order to fully understand the impact of the operating system, hyper-threading and virtualization on the CPU performances.

5 Summary

The HPC cluster NEMO has successfully been integrated into the workflow of local users in Freiburg running ATLAS data analysis jobs. This has been achieved in a transparent way using full virtualization on NEMO and utilizing the meta-scheduler ROCED for the integration of the virtual resources into the frontend scheduler. The system is in production since fall 2017.
First performance tests using different benchmarks show some degrading in performance of the virtual machines compared to running on bare metal. The differences are within the expected ranges when using different operating systems and get significantly reduced when going from pure CPU benchmarks like HS06 to benchmarks more closely related to high energy particle physics analysis like KV.
The continuous benchmarking effort will ensure a stable and efficient environment. Using the results of the benchmarks as input to the scheduling process will lead to an even more responsive system in the future.

6 Acknowledgements

The research is supported by the Ministerium für Wissenschaft, Forschung und Kunst Baden-Württemberg in the project MWK 34-7547.221 “ViCE Virtual Open Science Collaboration Environment” and by the Bundesministerium für Bildung und Forschung in the project 05H15VFCA1 “Higgs-Physik mit dem und Grid-Computing für das ATLAS-Experiment am LHC”.

References

[1] G. Apollinari, O. Bruening, T. Nakamoto, L. Rossi (2017), arXiv:1705.08830

[2] J. Shiers, Computer Physics Communications **177**, 219 (2007), proceedings of the Conference on Computational Physics 2006

[3] OpenStack Foundation, *OpenStack (Newton)* (2010), https://www.openstack.org/

[4] D. von Suchodoletz, B. Wiebelt, K. Meier, M. Janczyk, *Flexible HPC: bwForCluster NEMO*, in *Proceedings of the 3rd bwHPC-Symposium: Heidelberg 2016* (heiBOOKS, 2017), http://books.ub.uni-heidelberg.de/heibooks/reader/download/308/308-4-79237-1-10-20171002.pdf

[5] A.J. Gamel, U. Schnoor, K. Meier, F. Bührer, M. Schumacher (ATLAS Collaboration), Tech. Rep. ATL-SOFT-PROC-2017-070, CERN, Geneva (2017), https://cds.cern.ch/record/2292920

[6] M.A. Jette, A.B. Yoo, M. Grondona, *SLURM: Simple Linux Utility for Resource Management*, in *In Lecture Notes in Computer Science: Proceedings of Job Scheduling Strategies for Parallel Processing (JSSPP) 2003* (Springer-Verlag, 2002), pp. 44–60

[7] Adaptive Computing, *MOAB HPC SUITE* (2014), http://www.adaptivecomputing.com/products/hpc-products/moab-hpc-suite-grid-option/

[8] G. Erli, F. Fischer, G. Fleig, M. Giffels, T. Hauth, G. Quast, M. Schnepf, J. Heese, K. Leppert, J.A. de Pedro et al., Journal of Physics: Conference Series **898**, 052021 (2017)

[9] University of Wisconsin – Madison, *HTCondor* (2018), https://research.cs.wisc.edu/htcondor/

[10] P. Buncic, C. Aguado Sánchez, J. Blomer, A. Harutyunyan, M. Mudrinic, The European Physical Journal Plus **126**, 13 (2011)

[11] Fermilab and CERN, *Scientific Linux release 6.8 (Carbon)* (2011), http://www.scientificlinux.org/

[12] J. Blomer, G. Ganis, N. Hardi, R. Popescu, *Delivering LHC Software to HPC Compute Elements with CernVM-FS*, in *High Performance Computing*, edited by J.M. Kunkel, R. Yokota, M. Taufer, J. Shalf (Springer International Publishing, Cham, 2017), pp. 724–730, ISBN 978-3-319-67630-2

[13] A.P. Millar, G. Behrmann, C. Bernardt, P. Fuhrmann, D. Litvintsev, T. Mkrtchyan, A. Petersen, A. Rossi, K. Schwank, Journal of Physics: Conference Series **513**, 042033 (2014)

[14] The Fraunhofer-Institut für Techno- und Wirtschaftsmathematik ITWM, *BeeGFS* (2014), https://www.beegfs.io

[15] HashiCorp, *Packer 1.2.4* (2013), https://www.packer.io/

[16] Puppet, *puppet 3.8.7* (2005), https://puppet.com

[17] B. Blumenfeld, D. Dykstra, L. Lueking, E. Wicklund, Journal of Physics: Conference Series **119**, 072007 (2008)

[18] P. Buncic, C. Aguado-Sanchez, J. Blomer, A. Harutyunyan, Journal of Physics: Conference Series **331**, 052004 (2011)

[19] HEPiX Benchmarking Working Group, *HEP SPEC06 (HS06) benchmark* (2006), https://w3.hepix.org/benchmarking.html

[20] M. Alef, C. Cordeiro, A. De Salvo, A. Di Girolamo, L. Field, D. Giordano, M. Guerri, F.C. Schiavi, A. Wiebalck, J. Phys. Conf. Ser. **898**, 092056 (2017)

[21] R. Graciani, A. McNab, *Dirac benchmark 2012* (2012), gitlab.cern.ch/mcnab/dirac-benchmark/tree/master

[22] H. Curnow, B. Wichman, Computer Journal **19**, 43 (1976)

[23] A.D. Salvo, F. Brasolin, Journal of Physics: Conference Series **219**, 042037 (2010)

[24] P. Calafiura, W. Lavrijsen, C. Leggett, M. Marino, D. Quarrie (2005)

[25] The CentOS Project, *CentOS Linux release 7.4.1708 (Core)* (2017), https://www.centos.org/